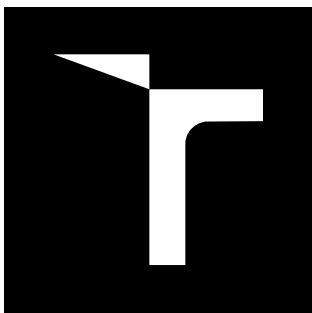


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta strojního inženýrství

BAKALÁŘSKÁ PRÁCE



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

**ÚSTAV MECHANIKY TĚLES,
MECHATRONIKY A BIOMECHANIKY**

INSTITUTE OF SOLIDMECHANICS, MECHATRONICS AND BIOMECHANICS

ŘÍZENÍ TŘÍOSÉHO MANIPULÁTORU

CONTROL OF THREE-AXIS MANIPULATOR

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Ján Rapčan

VEDOUcí PRÁCE

SUPERVISOR

Ing. Pavel Houška, Ph.D

BRNO 2017

Zadání bakalářské práce

Ústav: Ústav mechaniky těles, mechatroniky a biomechaniky
Student: **Ján Rapčan**
Studijní program: Aplikované vědy v inženýrství
Studijní obor: Mechatronika
Vedoucí práce: **Ing. Pavel Houška, Ph.D.**
Akademický rok: 2016/17

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Řízení tříosého manipulátoru

Stručná charakteristika problematiky úkolu:

Tato práce se zabývá návrhem a realizací řízení tříosého manipulátoru pro přesné polohování. Manipulátor je poháněn krokovými motory a základní senzorickou výbavu tvoří reференční a koncové snímače. Řízení bude realizováno na PLC s motorovými moduly firmy B&R.

Cíle bakalářské práce:

1. Seznamte se s programovacím prostředím AutomationStudio;
2. Seznamte se s použitými moduly pro řízení krokových motorů;
3. Provedte oživení jedné osy a změřte dosažitelné parametry jedné osy;
4. Realizujte řízení celého manipulátoru;
5. Provedte ověření a zhodnocení přesnosti polohování anipulátoru.

Seznam literatury:

Shell, R.L., Hall, E.L.: Handbook of industrial automation. New York: M. Dekker, c2000, xii, 900p. ISBN 0-8247-0373-1.

Pavelka, J., Čeřovský, Z., Javůrek, J.: Elektrické pohony, Nakladatelství ČVUT, Praha 2003

dokumentace PLC a modulů [online] Dostupná na <http://www.br-automation.com>

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2016/17

V Brně, dne

L.S

prof. Ing. Jindřich Petruška, CSc.
ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
děkan

ABSTRAKT

Práca sa zaoberá riadením a presným polohovaním dopravníka v 3D priestore. Pre riadenie polohovania je využité hardvérové a softvérové vybavenie od spoločnosti Bernecker&Reiner. Medzi ciele práce patrí aj oboznámenie sa z programovacím prostredím AUTOMATION STUDIO, v ktorom prebieha celá práca.

KLÚČOVÉ SLOVÁ

Polohovanie, Riadenie, PLC, X20SM1426

ABSTRACT

This thesis deal swith controlling and precision positioning of conveyor in 3D space. Equipment for controling of positioning is made by Bernecker&Reinercompany. Important objective of work is also familiarization with programing environment AUTOMATION STUDIO, which powers whole operation.

KEYWORDS

Positioning, Control, PLC, X20SM1426

BIBLIOGRAFICKÁ CITÁCIA

RAPČAN, J. *Řízení tříosého manipulátoru*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2017. 41 s. Vedoucí bakalářské práce Ing. Pavel Houška, Ph.D..

PREHLÁSENIE

Prehlasujem, že som túto bakalársku prácu na tému *Řízení tříosého manipulátoru* spracoval samostatne pod vedením Ing. Pavel Houška, Ph.D. a v zozname zdrojov uviedol všetky zdroje, z ktorých som čerpal.

V Brne dňa

.....
Ján Rapčan

POĎAKOVANIE

Rád by som na tomto mieste poďakoval svojmu vedúcemu práce Ing.Pavlovi Houškovi, Ph.D. za odborné rady, podnetné pripomienky, trpezlivosť a priateľský prístup pri tvorbe mojej bakalárskej práce.

OBSAH

OBSAH.....	9
1 ÚVOD.....	10
2 PLC.....	11
3 CIELE PRÁCE.....	15
3.1 PLC: X20CP1484 B&R.....	15
3.1.1 PARAMETRE	16
3.1.2 NAPÁJANIE PLC	17
3.1.3 LED UKAZOVATEĽ.....	17
3.2 MODUL PRE RIADENIA KRKOVÉHO MOTORA, X20SM 1426.....	18
3.3 AUTOMATION STUDIO.....	19
3.3.1 PRACOVNÉ PROSTREDIE	19
3.3.2 POPIS PRACOVÉHO PROSTREDIA	20
4 MOTION.....	23
4.1 MECHATRONICKÉ RIEŠENIE POHONOV.....	24
4.2 ZÁKLADNÉ POŽIADAVKY POHONNÝCH SYSTÉMOV.....	25
5 PRAKTICKÁ ČASŤ.....	26
5.1 VYTvorenie nového projektu.....	26
5.2 PROGRAMOVANIE JEDNEJ OSI	29
5.2.1 ACP10_MC - PLCopen MOTION CONTROL LIBRARY	30
5.3 PROGRAMOVANIE CELÉHO MANIPULÁTORA.....	34
6 ZÁVER	38
7 ZOZNAM POUŽITÝCH ZDROJOV.....	39
8 ZOZNAM POUŽITÝCH SKRATIEK A SYMBOLOV.....	40
9 ZOZNAM OBRÁZKOV A TABULIEK	41

1 ÚVOD

V súčasnej dobe takmer každý stroj alebo zariadenie v sebe obsahuje pohyblivé časti, ktoré je potrebné riadiť na základe špecifických požiadaviek. Riadenie pohonov, ktoré bolo v minulosti dosť komplikované pretože bolo výhradne založené na mechanických konštrukciách, teraz možno vykonávať s najvyššou mierou flexibility a efektivity s využitím najnovších technológií z oblasti riadenia pohonov prostredníctvom PLC¹. Optimálna vzájomná koordinácia jednotlivých zložiek systému pre riadenie pohonov, prináša značné výhody v oblasti riadenia technologických procesov. Siet' takto integrovaných častí v procese automatizácie tvorí jednu funkčnú jednotku pre riadenie pohonov. To nám dáva predpoklady zamerať sa predovšetkým na optimalizáciu technologického procesu nielen z hľadiska jeho presnosti, ale aj rýchlosti.

Moderným riešením v automatizácii je využívanie PLC. Tu sa naskytuje otázka čo to je a k čomu to slúži. PLC je priemyselný počítač ktorý pracuje v reálnom čase. Je charakteristický tým, že program vykonáva v cykloch. Nie je to nič zvláštne čo by nezvládol aj PC. Rozdielom oproti PC je, že priamo jeho periférie sú prispôsobené na napájanie technologických procesov.

Periférie sú v podstate vstupy a výstupy analógové alebo digitálne a taktiež množstvo prídavných modulov, ktoré firmy ponúkajú pre svoje programovateľné automaty.

Na trhu je niekoľko firiem, ktoré ponúkajú programovateľne logické automaty, rozširiteľné o moduly klasické digitálne alebo analógové, ale taktiež aj špeciálne moduly pre riadenie pohonov, hydraulických a pneumatických systémov.

Nás najviac zaujímajú moduly pre riadenie pohonov (Motion control). Podľa typu motora ktorí chceme riadiť musíme zvoliť správny typ modulu. Výkonnejšie motory a teda aj moduly pre ne určené potrebujú externé napájanie. PLC sa v tomto prípade používa pre ovládanie silovej časti pohonu.

¹ PLC – Programovateľný logický automat z anglického *ProgrammableLogicController*

2 PLC

Pre PLC sa používajú rôzne označenia: [7]

- PLC - Programmable Logic Controller.
- PC - Programmable Controller.
- SPS - Speicher Programmierbare Steuerung.
- FPC - Free Programmable Controller.
- PA - Programovateľný automat.

Prvé PLC [7] (sedemdesiate roky) malo jednoduchý systém, ktorý umožňoval pripojenie a spracovanie binárnych signálov. Zvládalo len logické operácie. Na túto činnosť používalo jednobitový procesor. Preprogramovalo sa v grafickom jazyku.

Typy PLC podľa prevedenia: [6]

- Kompaktné – všetko v jednom zariadení.
- Modulárne – prídavné moduly alebo zásuvné karty.
- So zabudovaným operátorským panelom – OPLC alebo pracovná stanica.

Podľa veľkosti : [4]

- Mikro PLC – do 20 vstupov / výstupov.
- Malé PLC – do 128 vstupov / výstupov.
- Stredná trieda – 128 až 512 vstupov / výstupov.
- Najväčšia trieda – až tisícky vstupov / výstupov.

Formálna definícia PLC pochádza z národnej elektrikárskej asociácie výrobcov NEMA: elektronický digitálny operačný systém určený na použitie v priemyselnom prostredí, ktorý používa programovateľnú pamäť pre vnútorné ukladanie užívateľských inštrukcií pre vykonávanie funkcií, ako je sekvenčná logika, časovanie, počítanie a aritmetika. Ovládanie prebieha pomocou digitálnych alebo analógových vstupov a výstupov. Vďaka tomu dokážeme ovládať rôzne typy strojov alebo procesov. Periférie prislúchajúce PC sú navrhnuté tak aby sa dali ľahko integrovať do priemyselného riadiaceho systému a tým pádom ich v ňom využívať.

Programovateľný automat (PLC) je počítač s operačným systémom a softvérom pre riadenie. Tento operačný systém je špecializovaný a optimalizovaný tak, aby spracovával prichádzajúce informácie v reálnom čase, teda v čase ich vzniku.[1]

Základná hardvérová architektúra PLC sa skladá z procesora, napájania, vstupov a výstupov. Procesor sa skladá z centrálnej procesorovej jednotky (CPU) a pamäte. CPU

pozostáva okrem mikroprocesora aj z rozhrania pre ovládanie vstup a výstupov (I/O) a ďalších komunikačných pripojení. Napájací zdroj PLC je obvykle samostatný modul, rovnako ako vstupy a výstupy, tieto moduly sú oddelené od procesora. [9]

Rôzne typy modulov I/O medzi ktoré patria: [4]

- Binárne digitálne vstupy (DI) – pre jednosmerné napájanie (5V, 12V, 24V, 48V), striedavé napájanie (24V, 48V, 115V, 230V). Sú usporiadané do skupín po 4, 8, 16, 32.
- Binárne digitálne výstupy (DO) – pre jednosmerné spínacie napájanie (24V, 48V), pre striedavé spínacie napájanie (24 až 250V).
- Kombinované moduly binárnych vstupov a výstupov.
- Analógové vstupy (AI) – pre pripojenie senzorov teploty, vlhkosti, tlaku, sily, polohy a rýchlosti. Potrebujú A/D prevodník so šírkou 8 alebo 12 bitov. Pre termočlánky a odporové teplomery sa používajú špecializované typy.
- Analógové výstupy (AO) – pre ovládanie akčných členov či zariadení so spojitým vstupným signálom (servopohony, frekvenčné meniče a ručičkové meracie prístroje). Potrebujú D/A prevodník so šírkou 8 alebo 12 bitov. Napäťové a prúdové. Prúdové: akčné (napájanie z PLC), pasívne (externé napájanie).
- Čítačové moduly – pre čítanie pulzov. Pre inkrementálne snímače polohy, snímače s pulzným alebo frekvenčným výstupom.
- Polohovacie moduly – pre snímanie a riadenie polohy.
- Špeciálne moduly – pre regulátory, pre aplikáciu fuzzy logiky a fuzzy regulácie, moduly pneumatických výstupov, moduly pre CCD kamery.
- Komunikačné moduly – pre komunikáciu medzi vzdialenými I/O modulmi alebo ďalším PLC a PC.

Spoločnosť B&R poskytujúca hardvér a softvér pre túto prácu ponúka niekoľko modulov pre riadenie pohonov [1]:

- ACOPOSmicro - tento modulový rad je určený pre riadenie motorov s nízkou spotrebou elektrickej energie. Ponúka širokú škálu menovitého napätia od 18V do 95V jednosmerného napätia. To umožňuje riadenia motorov s rôznym rozsahom napätia. ACOPOSmicro zvládne pripojenie dvoch motorov naraz. Je konštruovaný pre riadenie krokových motorov, s menovitým prúdom až do 10A a špičkovým prúdom max. 15A. A pre servomotory s nominálnym prúdom do 8A a špičkovým do 15A. Tento model v sebe zahŕňa výstupy z napájaním 24V ktoré je možné využiť pre zapojenie externej brzdy.
- ACOPOS – tento model je určený pre inteligentné servopohony. Táto modelová rada je výnimočná svojou pracovnou rýchlosťou, 50 ms v regulačnej slučke. Veľká pozornosť bola venovaná aj odolnosti celého modulu voči nepriaznivým

pracovným podmienkam. Modul vie prečítať mikročip motora a zapísať hodnoty do konfigurácie bez zásahu programátora (ak je motor takýmto čipom vybavený)

- ACOPOS P3 – je určený pre 3-osé servopohony s vysokou presnosťou polohovania a dynamikou systému. Má integrované bezpečnostné prvky. Kvôli jeho rýchlemu reakčnému času musí byť pripojený k PLC pomocou dátovej zbernice ethernet powerlink².
- ACOPOSmulti – určený pre viac ose polohovanie. Generácia s vysokým stupňom ochrany. Je vybavený ventilátorom pre odvod tepla v kombinácii s možnosťou pripojenia chladiaceho systému na báze vody alebo oleja. Ponúka možnosť polohovania z miesta na miesto, CNC alebo elektrickú kompenzáciu prevodov.
- ACOPOSmotor – je kombináciou elektromotora s integrovaným systémom polohovania a taktiež príslušnej senzoriky. Ponúka sa v rozsahu krútiacich momentov od 1,2 do 10 Nm a výkonovom rozsahu od 550 W do 2,3 kW. Spojenie s hlavným riadiacim PLC je riešené pomocou hybridných káblov. Tento kombinovaný modul je vybavený stupňom ochrany IP65.
- ACOPOSinverter – kombinuje frekvenčný menič spolu s modulom určeným pre riadenie trojfázových asynchrónnych a synchronných motorov s napájacím napätím od 200 do 500 V vo výkonnostnom rozsahu od 0,18 do 15 kW. Je vybavený funkciami pre rozloženie zaťaženia, obmedzenie krútiaceho momentu a PID regulátorom.

Norma IEC 61131 definuje model pamäte a programu podľa najnovších softvérových trendov. Tento model zahŕňa vlastnosti dizajnu, štruktúrovaného programovania, hierarchickej organizácie a formálne softvérové rozhranie [2].

Konfigurácia je telom softvéru (program a dáta), ktoré prislúchajú systému PLC. Všeobecnou zásadou je, že konfigurácia a program prislúcha len jednému PLC. Vo veľkých komplexných systémoch ktoré vyžadujú použitie viac ako jedného PLC je ich vzájomná komunikácia definovaná na základe rozhraní, takzvaných prístupových ciest.

PLC program beží na operačnom systéme ktorí sa označuje ako Runtime [10].

Runtime je operačný systém ktorí tvorí jadro softvéru ktoré umožňuje programu fungovať. Má modulárnu štruktúru so schopnosťou rýchlo vykonávať program v presnom časovom rámci. To zaručuje kvalitu a presnosť. Runtime je plne integrovaný do modulov a komponentov spoločnosti B&R to umožňuje prístup aplikačnému programu k I/O, komunikačnému rozhraniu a ukladaniu dát. Ponúka rozsiahle knižnice a funkcií v súlade s IEC 61131. Poskytuje prístup k všetkým sieťam a zberniciam. Obsahuje VNC klienta pre simulačný mód.

² Ethernet powerlink protokol umožňuje prenos údajov v cykle do 200 µs a ultrarýchly prenos do 1 µs.

PLC

Program spravidla pozostáva z prepojenia funkčných blokov. Každý z funkčných blokov môže byť napísaný v ktoromkoľvek jazyku, ktorý splňuje normu IEC 61131-3. Program okrem funkčných blokov obsahuje lokálne premenné a fyzické pripojenie s I/O. Program je schopný čítať a zapisovať globálne premenne na I/O a komunikovať s ostatnými programami. Premenné sú deklarované v rámci rôznych softvérových prvkov. [5]

Komunikácia medzi PLC, PC, rozširujúcimi modulmi a riadiacimi jednotkami pohonov [3]:

- Ethernet (ethernet powerlink, Modbus TCP/IP, Ethernet/IP, PROFINET, ..). [11]
- Profibus, Modbus RTU, Interbus, IO Link. [4]
- CAN (CANopen, DeviceNet). [12]

PLC sa využíva v automatizácii priemyselných elektromechanických procesoch, na kontrolu strojov, v montážnych linkách, v zábavných dráhach alebo svetidlách. Automobilový priemysel je stále jedným z najväčších používateľov PLC. Ako náhle sa PLC program spustí beží kontinuálne ako slučka na dobu neurčitú.

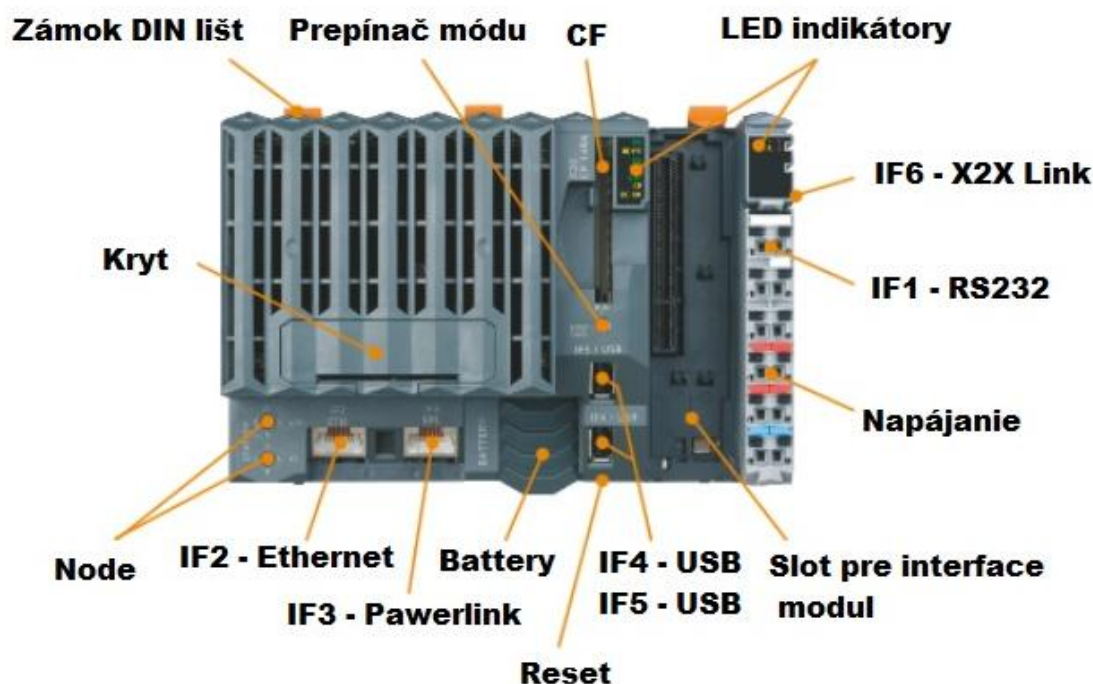
Dokážu pracovať v rôznom rozsahu teplôt, a sú odolné voči elektrickému rušeniu a taktiež vibráciám, nárazom, nečistotám, vode, ochrana IP67.

3 CIELE PRÁCE

Cieľom práce je oboznámenie sa s problematikou riadenia pohonov pomocou PLC. V práci je využívaný softvér a hardware od spoločnosti Bernecker&Reiner. Na dosiahnutie cieľového zadania pre riadenie trojosového dopravníka, bolo nutné oboznámiť sa so softvérom Automation studio, ktorý je určený k programovaniu PLC od spoločnosti B&R. K práci boli využité tieto komponenty: PLC: X20CP1484, 3x modul pre riadenie krokových motorov X20SM1426, 3 x krokový motor.

3.1 PLC: X20CP1484 B&R

CPU s označením X20 je optimálny model pre širokú škálu potrieb. Táto modelová rada zvláda časové cykly do 100ms. V základnom pripojovacom rozhraní je ethernet, USB a ethernet powerlink. Pre náročnejšie aplikácie sa môže rozšíriť o interface³ modul. Je schopný zvládnuť pripojenie 250 I/O (3000 kanálov) modulov a preto sa zaraďuje do strednej triedy CPU. Využíva pasívne chladenie.



Obr. 3.1 Popis PLC firmy B&R.[1]

³ Rozširujúci model pre špecifické aplikácie. Je vybavený komunikačným rozhraním RS232 (sériový port). RS232 slúži k sériovej komunikácii medzi dvoma zariadeniami. Sériová komunikácia to znamená, že jednotlivé bity prenesených dát sú vyslané postupne za sebou (v sérii).

3.1.1 PARAMETRE

X20 CPU, Celeron 266 kompatibilný 32 MB DRAM, 1 MB SRAM, vymeniteľná aplikačná pamäť: CompactFlash⁴, 1x slot pre moduly X20, 2x USB rozhranie, 1x rozhranie RS232, 1x rozhranie Ethernet 10/100 Base-T, 1xPowerlink V1 / V2 rozhranie, modul napájacieho zdroja, TB12 svorkovnica a vymeniteľný kryt. [1]

Programy pre ovládanie stroja sú uložené v statickej pamäti RAM, ktorá je napájaná batériou.

Napájanie CPU a X2X⁵link.

Vstupné napätie: 24VDC -15% / +20%.

Vstupný prúd: Max.2.2A.

Tabuľka 3.1 Rozhrania prislúchajúce PLC X20CP1484. [1]

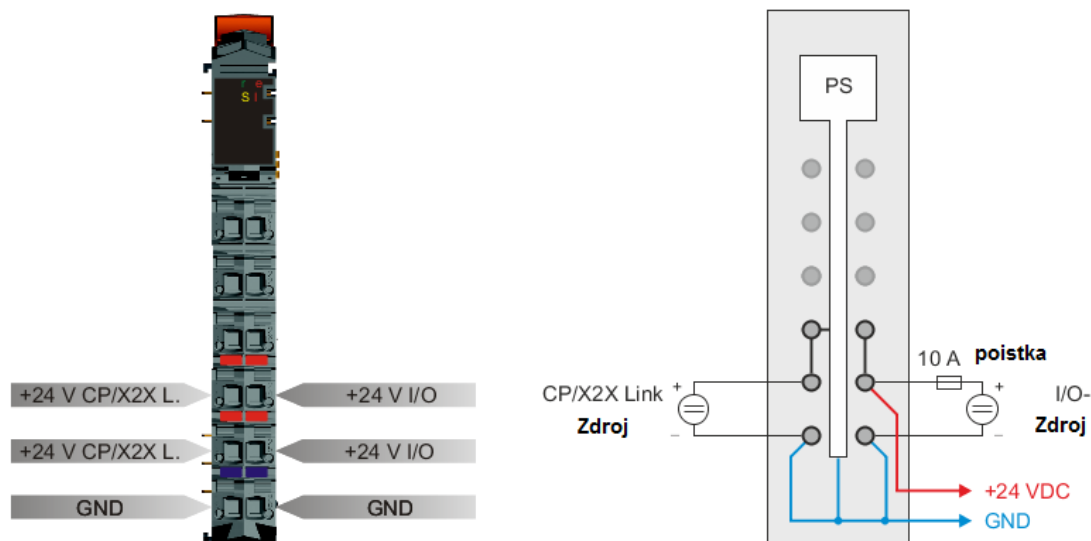
Rozhranie IF1	
Typ	RS232
Dizajn	Spojenie s 12-pinovov zbernicou TB12
Prenosová rýchlosť	Max. 115.2 kbit/s
Rozhranie IF2	
Typ	Ethernet
Dizajn	RJ45
Prenosová rýchlosť	10/100 Mbit/s
Dĺžka	Max. 100 m medzi dvomi stanicami
Rozhranie IF3	
Zbernica	POWERLINK V1/V2 (komunikácia v reálnom čase)
Typ	100 Base-T (ANSI/IEEE 802.3)
Dizajn	RJ45
Prenosová rýchlosť	100 Mbit/s
Dĺžka	Max. 100 medzi dvomi stanicami
Rozhranie IF4/IF5	
Typ	USB 1.1
Dizajn	Typ A
Množstvo	2
Rozhranie IF6	
Typ	X2X Link
Množstvo	1

⁴CompactFlash - (CF): pamäťová karta

⁵X2X link - Je komunikácia medzi PLC a I/O modulmi


3.1.2 NAPÁJANIE PLC

Na nasledujúcom obr.3.2 je zobrazený postup ako správne pripojiť napájanie pre PLC a prídavné moduly, ktoré majú napájanie na spoločnej zbernici.



Obr. 3.2 Popis zapojenia napájania pre PLC. [1]

3.1.3 LED UKAZOVATEĽ

Obrazok	LED	farba	Status	Popis
	r	zelená	Vypnutá	Zdroj nie je pripojený
			Jedno bliknutie	Reset
			Bliká	Prípravný mód
			Zapnutá	RUN mód
	e	červená	Vypnutá	Zdroj nie je pripojený
			Dvojité bliknutie	LED indikuje jeden zo statusov:
				<ul style="list-style-type: none"> X2X Link nefunguje I/O príliš nízke napätie výstupne napätie X2X Link je malé
	S	žltá	Vypnutá	RS232 neaktívne
			Zapnutá	Preposielanie dát cez RS232 rozhranie
	I	červená	Vypnutá	X2X Link je v poriadku
			Zapnutá	X2X Link, preťažený zdroj

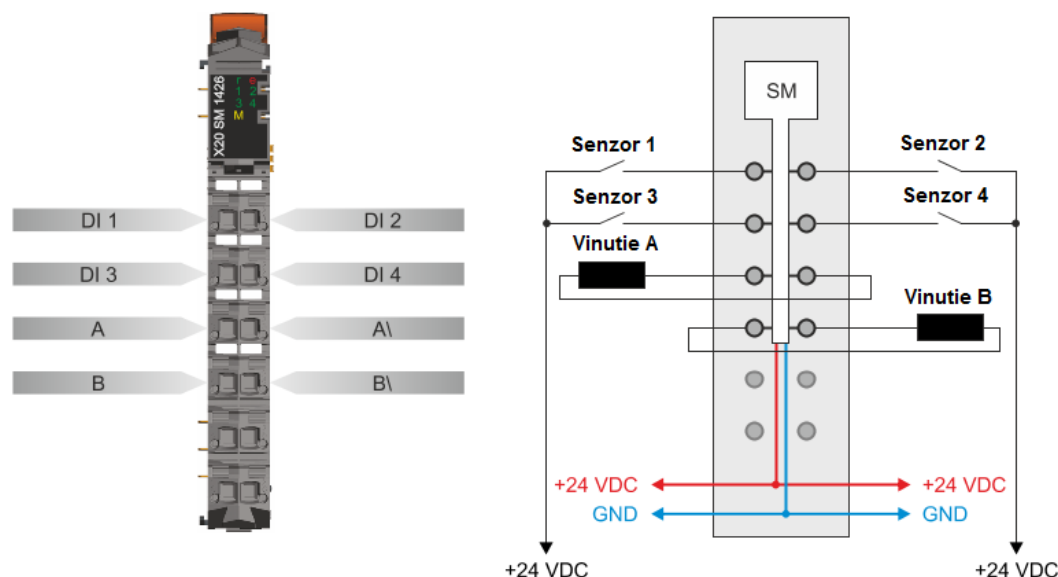
Obr. 3.3 Význam jednotlivých kontroliek na LED panely.[1]

3.2 MODUL PRE RIADENIA KROKOVÉHO MOTORA, X20SM 1426

Modul slúži na ovládanie krokových motorov s menovitým napätím 24V DC pri prúde až do 1A (1,2A vrchol). Navyše tento modul má štyri digitálne vstupy, ktoré môžu byť použité ako koncové spínače alebo ako vstupy snímačov. Individuálnym nastavením cievok je motor prevádzkovaný iba s prúdom ktorý skutočne potrebuje. To zjednodušuje výber z dostupných motorov a zabráňuje zbytočným stratám ktoré sa maria na teplo. Tým znižujeme spotrebu elektrickej energie a tepelné zaťaženie, účinky sú pozitívne na životnosť celého systému. Kompletná flexibilita je dosiahnutá použitím hodnoty pre skutočný prúd, maximálny prúd a menovitý prúd, ktoré sú úplne nezávislé na sebe.

Systém automaticky identifikuje motor, čo je veľká pomoc počas stavu nečinnosti. Modul dokáže identifikovať pripojenie motora a vytvárať spätnú väzbu v podobe analógovej hodnoty. To umožňuje detekovať nielen chyby zapojenia, ale tiež nesprávne typy motorov. Mechanizmus detekcie je integrovaný a dokáže analyzovať zaťaženie motora. To umožňuje zastavenie motora v prípade preťaženia.

Modul je napájaný zo spoločnej zbernice ako PLC. Krokové motory sa pripoja na svorky A až B/. DI1 až DI4 sú digitálne vstupy na ktoré sa môžu pripojiť snímače podľa obr. 3.4.



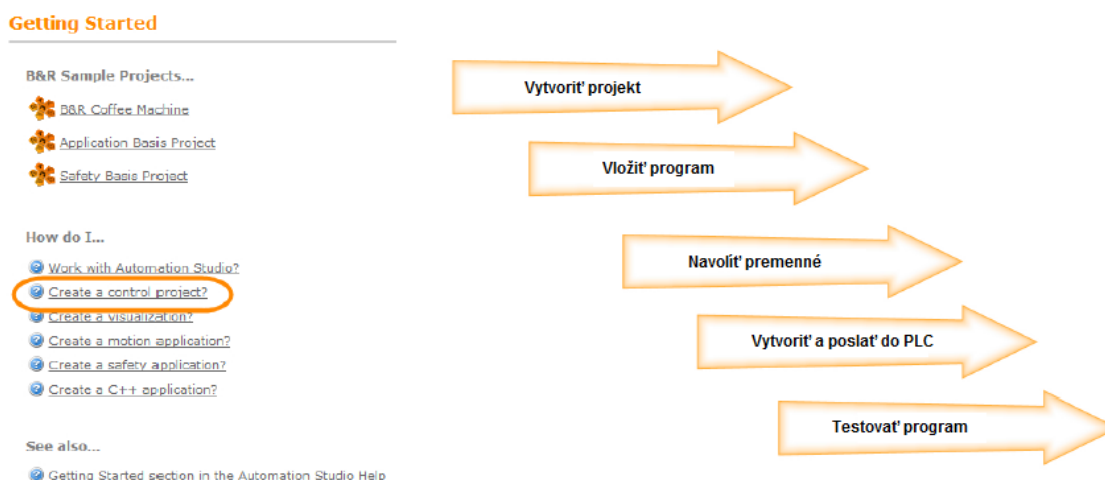
Obr. 3.4 Príklad zapojenia modulu s krokovým motorom a senzormi. [1]

3.3 AUTOMATION STUDIO

Je programové prostredie vytvorené firmou B&R, jeho výhodou je, že všetko potrebné sa nachádza priamo v ňom. Ak potrebujeme pre danú aplikáciu vytvoriť vizualizáciu, alebo riadenie pohonov všetky potrebné komponenty a knižnice sú zakomponované v automation studio. Výborným nástrojom je taktiež Help, ktorý obsahuje príklady využitia rôznych funkcií ako aj hardvérového príslušenstva.

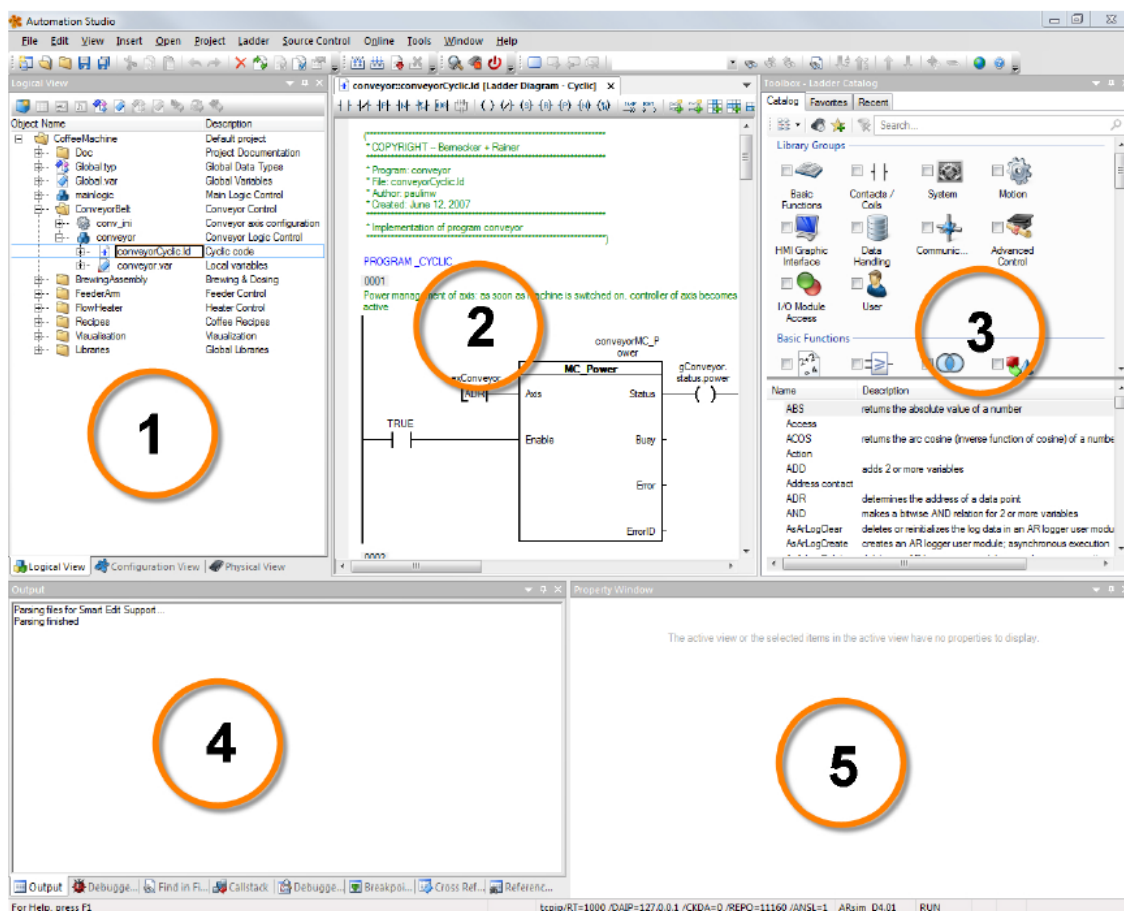
3.3.1 PRACOVNÉ PROSTREDIE

Na začiatku sa otvorí úvodné okno programovacieho prostredia na ktorom nájdeme rýchle odkazy pre vytvorenie ovládacieho programu, vizualizácie, pohybovej aplikácie a bezpečnostnej aplikácie. V úvode sú zakomponované aj tri jednoduché projekty pre predstavu. Takmer každý problém, s ktorým sa počas vytvárania vlastnej aplikácie stretneme je už vyriešený v helpe s názornou ukážkou ako postupovať v jeho odstránení. Ak sa nám nepodarí problém svojpomocne s použitím helpu odstrániť, stále máme možnosť obrátiť sa na on-line podporu ktorú firma B&R poskytuje svojim zákazníkom, či už telefonicky alebo prostredníctvom e-mailu.



Obr. 3.5 Úvodná obrazovka s diagramom krokov pri práci. [1]

3.3.2 POPIS PRACOVNÉHO PROSTREDIA



Obr. 3.6 Pracovné prostredie. [1]

1. Projektový prehliadač sa nachádza na ľavej strane okna. Ten sa používa na spravovanie a editáciu softvéru a konfiguráciu objektov v projekte. Táto časť obsahuje tri záložky:
 - Logical View – v tejto záložke sú umiestnené všetky programové záležitosti. Medzi ktoré patrí deklarácia premenných (globálnych/lokálnych), samotná programová časť ktorá sa môže skladať s viacerých podprogramov. Ich vzájomná komunikácia vzniká na základe využitia globálnych premenných.
 - Physical View – táto časť obsahuje strom konfigurácie zariadení, pričom na vrchole je typ použitého CPU. Následne sú pod záložkami rozhraní (X2X link, USB, Ethernet Powerlink) aktuálne pripojené moduly a zariadenia s ktorými sa pracuje.
 - Configuration View - táto časť je určená pre konfiguráciu Motion, spúšťanie testovacieho módu , rozloženie programov v cyklických triedach a samotné mapovanie premenných na I/O

2. Na tomto mieste sa nachádza editor zdrojového kódu. S funkciou záložiek kedy môžeme preskakovať s hlavného programu do vedľajšieho.
3. Okno Toolbox sa nachádza na pravej strane okna. V ňom sú obsiahnuté hardvérové a programové moduly ktoré potrebujeme pri zakladaní nového projektu. Vyber sa potvrdí dvojklikom alebo pretiahnutím do sekcie Configuration View. Prekliknutím na záložku Logical View sa zmení obsah Toolboxu. Zobrazia sa v ňom funkcie a softvérové objekty (programy podľa typu jazyka v ktorom sa budú písať) ich potvrdením sa vložia do projektu na ktorom sa aktuálne pracuje.
4. Okno výstupov je umiestnený na spodnej ľavej časti hlavného okna. Používa sa na zobrazenie informácií, ako sú správy, ktoré sú generované, keď sa projekt buduje. Slúži ako informačné okno ktoré nás upozorňuje na chyby pri nahrávaní do PLC (chýbajúce zátvorky, nedeklarovanú premennú a podobne).
5. Okno Vlastnosti sa nachádza v pravom dolnom rohu. Takisto sa tu zobrazuje možnosť konfigurácie. Podľa toho, aký objekt alebo hardvérový modul je aktuálne vybratý. Môže byť tiež využité k úprave vlastností označeného objektu v tomto okne.

Programovanie:

Editor zdrojového kódu podporuje programovanie v jazykoch podľa normy IEC 61131-3. Táto norma zahŕňa šesť programovacích jazykov. Automation studio rozširuje túto ponuku o programovací jazyk ANSI C/C++ .

Systém diagnostik manažér SDM je súčasťou programového balíka obsiahnutého v Automationstudio. Je to diagnostický nástroj. Jeho obsahom je aj Checklist, v ktorom sa zaznamenávajú všetky varovania pred závažnejším problémom. Checklist je súčasťou Loggeru ktorý beží spolu s programom, ale je vysunutý do menej podstatných cyklických tried. Ak by nám predvolené nastavenie vo štvrtej cyklickej triede nevyhovovalo nie je problém ho posunúť až do ôsmej. Do Loggeru sa zapíše error spolu s jeho číslom a stručným popisom, pre dôkladnejšie zhodnotenie problému je nutné si pomocou čísla erroru vyhľadať v helpe následné riešenie.

SDM sa dá otvoriť aj vo webovom prehliadači ak je PLC pripojené na sieť pomocou ethernetu, preto je vhodné aby PLC-čka boli zasieťované pre jednoduchú a rýchlu diagnostiku z hocikákeho miesta s pripojením na internet. V internetovom prehliadači si môžeme pozrieť záznam z logbooku.

SDM obsahuje systémové informácie ako sú: Status baterky, teplota CPU, status na CPU, využitie CPU, systémový čas, informácie o využití pamäte(DRAM⁶, SRAM⁷),

⁶je pamäť RAM po vypnutí sa dáta s nej stratia

⁷statická pamäť RAM je napájaná baterkou dáta sú zachované aj po vypnutí

CIELE PRÁCE

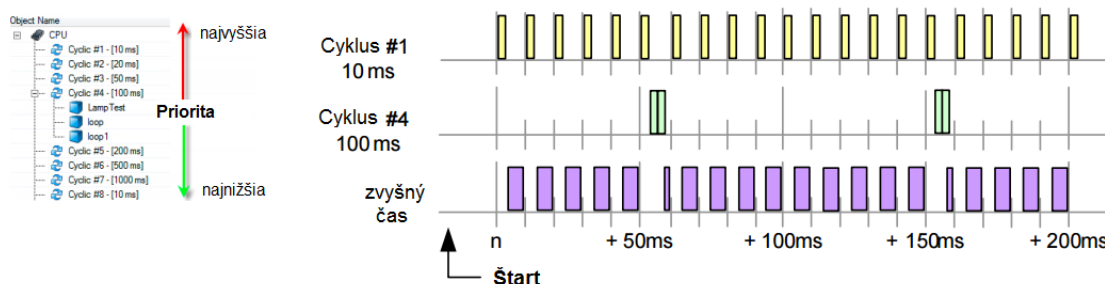
časovanie všetkých cyklov, vďaka čomu vieme zlepšiť využitie procesov na plno. Softvérové a hardvérové informácie.

RUNTIME:

Počas spúšťania Runtime sa na pozadí vykonáva:

- Kontrola hardvéru
- Aktualizácia hardvéru /firmvéru, ak je to nutné
- Kontrola všetkých modulov
- Moduly sa prekopírujú z ROM pamäte do DRAM
- Zachováva hodnoty premenných prekopírovaných do pamäte DRAM
- Nastaví variabilnú pamäť pre inicializáciu
- Aktivuje programy v cyklických triedach

Každá úloha má vopred definovanú dobu cyklu. Všetky povinnosti musia byť ukončené pred skončením cyklu. V prípade, že súčet doby prevádzky presahuje nakonfigurovanú dobu cyklu, PLC prejde do servisného módu. Riešením je pridať toleranciu k dobe cyklu alebo premiestniť program do nižšej cyklickej triedy. Doba cyklov je preddefinovaná pri vytváraní projektu, dá sa však editovať podľa potreby.



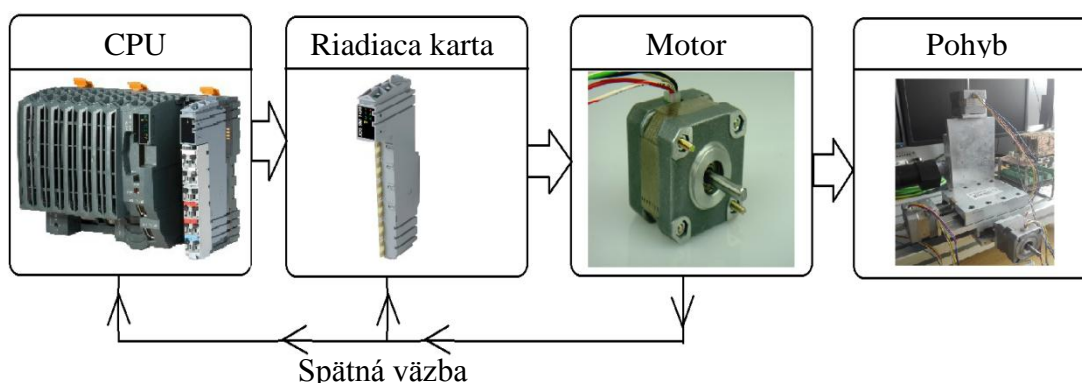
Obr. 3.7 Graf cyklov a umiestnenie podľa priority operácie. [1]

Zvyšný čas, je to čas medzi cyklami kedy aktuálne neprebíha žiadna operácia v programe. Táto doba je aj označovaná ako doba nečinnosti. Tento čas je následne využívaný na:

- On-line komunikáciu
- Aplikácie vizualizácie
- Prístup k súborom

4 MOTION

Problematika riadenia pohonov je v oblasti automatizácie a mechatroniky nazývaná ako Motion. Táto technológia sa používa pre uľahčenie automatizácie výroby medzi ktoré patrí riadenie elektromechanických, hydraulických a pneumatických pohonov. Pre motion sa tiež používajú výrazy ako servoriadenie alebo robotika.



Obr. 4.1 Schéma riadenia pohybu. [1]

Systém na obr. 4.1 je so spätnou väzbou, ktorý sa používa na riadenie polohy, rýchlosti a zrýchlenie. Spätná väzba obsahuje algoritmy pre uzatvorenie slučky (typicky pozície alebo rýchlosti), a tiež spracováva informácie zo vstupov / výstupov. Motor je poháňaný signálmi z riadiacej karty. Motor môže byť jednosmerný alebo striedavý, rotačný alebo lineárny. Motor je elektromagnetický ovládač, ktorý generuje požadované sily aby sa vykonával pohyb.

Účelom všetkých pohonov je presunúť nejaké zaťaženie. Spôsob, akým sa bremeno pohybuje, je známa ako profil pohybu. Profil pohybu môže byť tak jednoduché, ako pohyb z A do B po jednej osi, alebo to môže byť komplexné riadenie, v ktorom sa využíva viacero osí, ktoré sa musia pohybovať v zájomnej koordinácii. Existuje niekoľko typov pohybových profilov používaných s pohonnými systémami. Najčastejšie sa používajú so stálou rýchlosťou.

Hlavným faktorom je poloha a rýchlosť stroja. Tieto faktory sú kontrolované pomocou zariadení, ktoré sú pre túto činnosť priamo určené. Riadenie pohybu je dôležitou súčasťou robotiky a CNC obrábacích strojov, avšak riadenie týchto zariadení je oveľa komplikovanejšie kvôli kinematike celého systému. Riadenie pohybu je široko využívané v tlačiarenskom priemysle, pri výrobe elektronických súčiastok a montážnom priemysle. Motion zahŕňa všetky technológie súvisiace s pohybom objektov. Pokrýva všetky pohybové systémy od mikro až po veľké systémy, ako sú vesmírne platformy. Momentálne sa riadenie zameriava na riadenie systémov s elektrickými motormi ako sú DC/AC servomotory rovnako ako ovládanie robotických manipulátorov. Pretože väčšina z robotických manipulátorov je poháňaná elektrickými servomotorami ktorých hlavnou úlohou je kontrola pohybu.

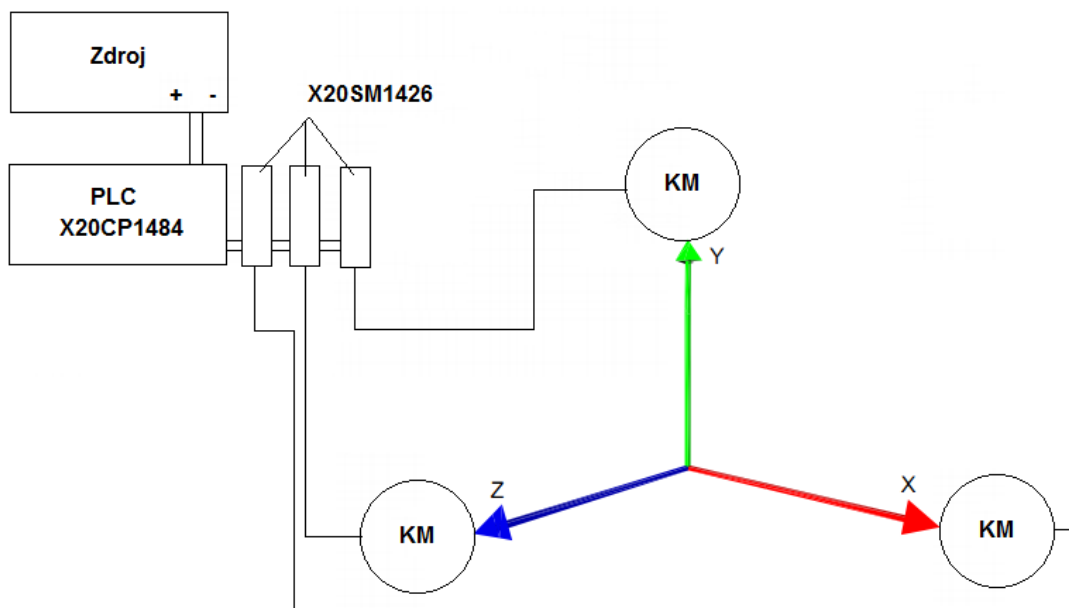
Trend sa jasne pohybuje v smere mechatronického riešenia pohonov. Pohybové sekvencie sú vykonávané za použitia mechanickej konštrukcie, ktoré sú v niektorých prípadoch dosť komplikované. Pohon je riešenie, ktoré je jednotné a môžu byť použité v rôznych systémoch, hrá dôležitú rolu v každom odvetví. Čím viac jednotlivých komponentov dokážeme koordinovať, tým silnejšia bude naša technológia. Mechatronické pohonné systémy môžu byť integrované do procesu ako funkčný celok. To umožňuje vývojárom zamerať sa predovšetkým na optimalizáciu procesu.

4.1 MECHATRONICKÉ RIEŠENIE POHONOV

Medzi podmienky spojené s mechatronickým riešením pohonov, patria elektrické pohony, prenos výkonu, proces konfigurácie pohonu a servopohony. Tieto a podobné výrazy sa často používajú na opis zložiek v systéme pohonov. K dispozícii je široká škála elektrických pohonov.

Nasledujúce otázky musia byť zodpovedané pred začiatkom každého projektu:

- Aké súčasti tvoria pohonný systém alebo polohovanie?
- Aké sú rozdiely medzi technológiami?
- Aké špecifikácie rozdeľujú technológie pre konkrétne použitie?



Obr. 4.2 Základné komponenty použitého systému.

Skladá sa z :

- DC zdroja 24V,
- PLC X20CP1484,
- 3x modul X20SM1426,
- 3x krokový motor,
- 3x mechanický sústava.

Zdroj napája PLC, v ktorom beží program na ovládanie manipulátora. Moduly X20SM1426 prenášajú požiadavky získane s PLC do krokových motorov, tie následne začnú vykonávať pohyb podľa parametrov ktoré prijali z PLC.

4.2 ZÁKLADNÉ POŽIADAVKY POHONNÝCH SYSTÉMOV

Systém musí byť veľmi dynamický, presný a poskytovať mimoriadnu mieru spoľahlivosti. Slovo "dynamika" je všeobecný pojem, ktorý zahŕňa silu, pohon alebo silu, ktorá riadi alebo na ne reaguje. Tým vzniká otázka ako meniť sily v priebehu času.

V praxi sú často nevyhnutné tieto charakteristiky: [13]

- Rýchle dosiahnutie požadovanej rýchlosti.
- Rýchle dosiahnutie presnej polohy.
- Byť schopný udržiavať určitú rýchlosť v čase.
- Byť schopný udržať vopred stanovený krútiaci moment.

V dôsledku toho, musí byť hnací systém schopný presne ovládať pripojené mechanické súčiastky v súlade so špecifikáciami - pri uplatňovaní najväčšieho množstva sily - bez straty presnosti. Táto vlastnosť priamo závisí od produktivity stroja. V mnohých aplikáciách je presnosť polohovania najdôležitejšia. Navyše jeho dynamické vlastnosti musia byť tiež schopné zachovať presnosť polohovania a regulovať podľa požiadaviek vhodné množstvo sily. Voľba elektromotora nie je jediným rozhodujúcim faktorom. Sofistikované meracie zariadenie a riadiace algoritmy tiež hrajú významnú úlohu pri manipulácii s týmito požiadavkami. Vysoké požiadavky možno splniť len vtedy, keď všetky komponenty v systéme vzájomne hladko spolupracujú.

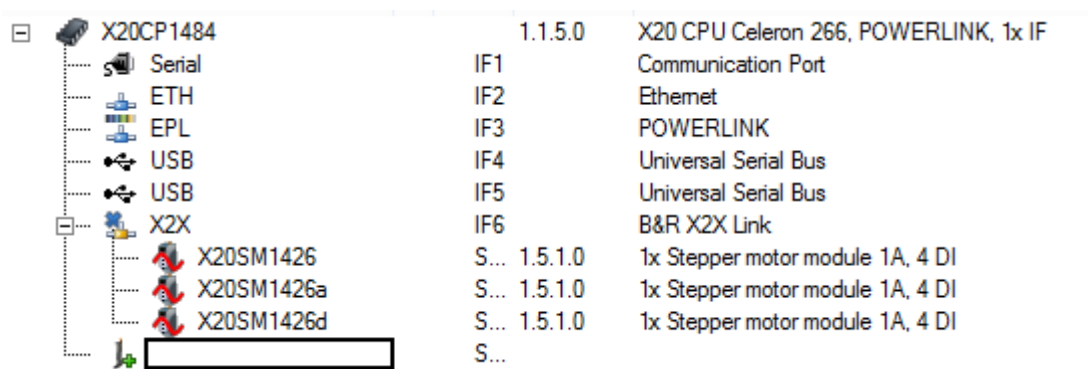
Pohonný systém je sústava tvorená mechanizmami a ďalšími súčastami, ktoré premieňajú, prenášajú alebo využívajú energiu, zaťaženie či pohyb ku stanoveným účelom. Každý stroj pozostáva zo vstupnej, premennej a výstupnej(pracovnej) časti. Hnací stroj (elektromotor) premieňa jeden druh energie na druhý. Pracovný stroj vykonáva zmenu tvaru, veľkosti, vlastnosti alebo polohy. Táto fyzická štruktúra podlieha zákonom o zmene energie.

Pri pohľade na sústavu, je najlepšie začať na stroji, ktorý je zvyčajne poháňaný elektromotorom. Elektrický motor mení elektrickú energiu na mechanickú energiu. To má za následok rôzne úrovne krútiaceho momentu a sily. Aby bolo možné kontrolovať tieto hodnoty podľa potreby, je potrebný riadiaci člen (PLC). Pre presné polohovanie je dôležité poznať aktuálnu pozíciu. V týchto prípadoch sa používajú senzory polohy, tie sú zvyčajne umiestnené priamo na motore. PLC spracuje všetky údaje. Akčný člen (elektrický motor) spustí svoju činnosť, pritom PLC vyhodnocuje údaje o polohe, rýchlosti a zrýchlení.

5 PRAKTICKÁ ČASŤ

5.1 VYTVORENIE NOVÉHO PROJEKTU

V Automation studiu vytvoríme nový projekt. Z pravého okna “Toolbox“ v záložke “PhysicalView“ vyberieme z produktov “Controller“ ktorý pre danú aplikáciu používame. V našej aplikácii je použitý Controller X20CP1484 ktorý sme si popísali v kapitole 3.1. K PLC je nutné priradiť všetky moduly, ktoré sú k nemu pripojené. Ak tak neurobíme PLC ostane v servisnom móde. Pred nahratím je potrebné manuálne nastaviť konfiguráciu ethernetového pripojenia. Je potreba aby PC a PLC mali rovnakú masku siete a rovnakú IP adresu.



Obr. 5.1 Strom pripojených zariadení.

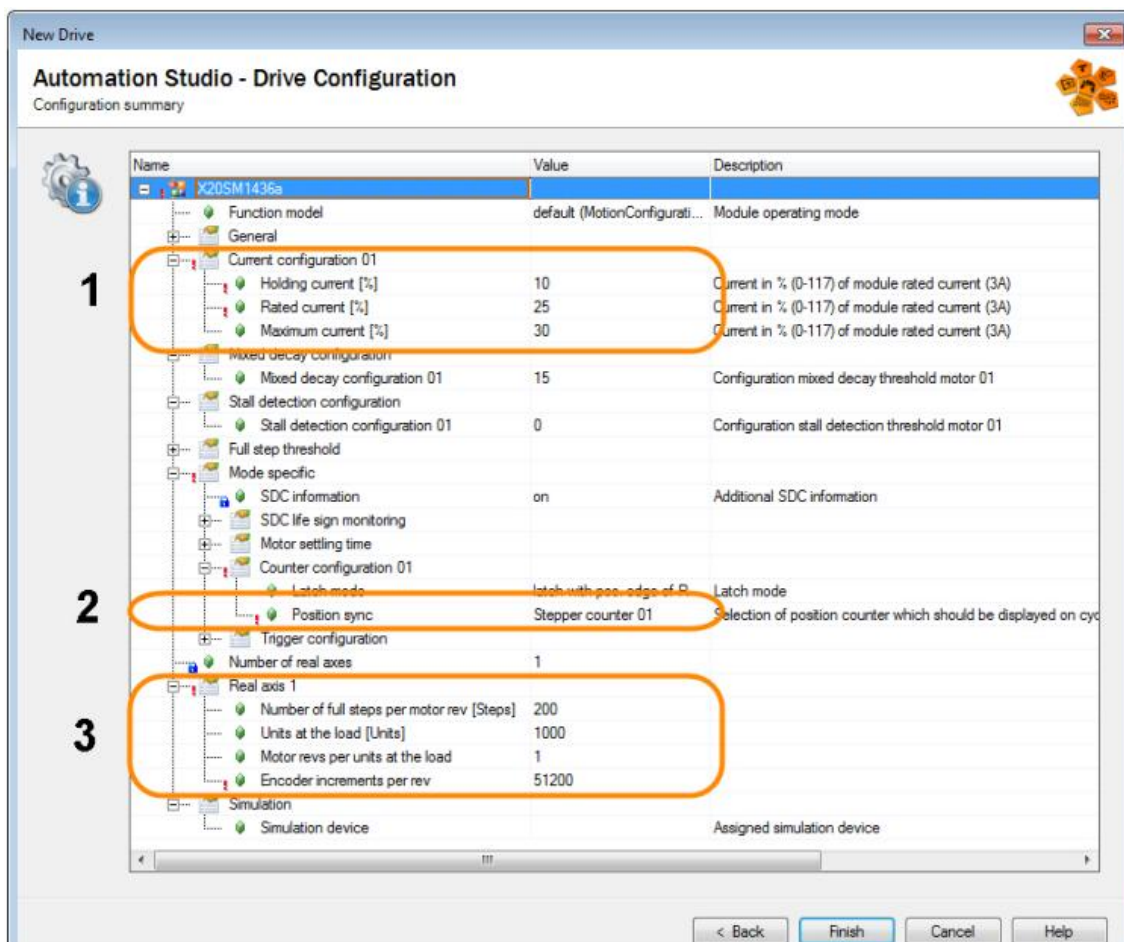


Obr. 5.2 Reálne zobrazenie PLC s pridanými modulmi (prevzaté so System Designer⁸).

⁸ Je súčasťou programového balíka Automation studio. Slúži ako grafický nástroj pre predstavu v akom poradí sú moduly pripojené k PLC.

PRAKTICKÁ ČASŤ

Po vložení modulov do Physical View, Automation studio spustí sprievodcu nastavenia nového pohonného modulu. Nastavenie je znázornené na obr. 5.3.



Obr. 5.3 Nastavenie parametrov motora.

1. Konfigurácia prúdu:

Prúd musí byť nakonfigurovaný správne⁹ aby sme motor rozbehli ale nepreťažili. Zádržný, nominálny a maximálny prúd sa nastavujú tu, pričom sa zadáva hodnota prúdu v percentách. Modul je schopný dodať 1A prúdu tj.100%. Vysoký zádržný prúd spôsobí zahrievanie aj keď motor stojí. Ak je nominálny prúd príliš nízky, motor sa nemusí rozbehnúť alebo nedosiahne požadovanú rýchlosť

2. Spätná väzba od polohy:

Motor môže byť použitý buď s alebo bez spätnej väzby. V konfigurácii ktorú si popíšeme je príklad bez snímača so spätnou väzbou. V tomto prípade je hodnota pre parameter „Positionsync“ nastavená takto: „Stepper Counter01“. Skutočná poloha motora sa potom musí vypočítavať interne.

⁹Pracovný bod motora by mal byť zvolený tak, aby prúd bol menší než alebo rovný nominálnej hodnote prúdu. Nesprávna konfigurácia môže spôsobiť, že motor bude preťažený. Toto nastavenie sa môže líšiť v závislosti na aplikácii.

3. Tri jednotky v rámci rotácie motora:

Krokové motory pracujú v mikrokrokoch, každý riadny krok má rozlíšenie 256MicroStep, ak rotácia motora trvá 200 plných krokov, potom urobí 51 200 mikrokrokov. Táto hodnota musí byť nakonfigurovaná pre parameter „Encoderincrements per revolution“ ak nie je použitý externý senzor. Systém automaticky prevedie jednotky tak, aby zodpovedali zaťaženiu. V uvedenom príklade 1000 jednotiek (PLCunits) zodpovedá 51 200 mikrokrokom, čo je rovnaké ako jedna otáčka motora.

Úrovně stop a koncové spínače sa budú konfigurovať neskôr. V prípade, že logika vstupov je nesprávna alebo vstupy neboli pevné, potom sa zabráni pohybu osi.

Parametre, ktoré boli nakonfigurované v sprievodcovi potom môže byť vyvolaná a opätovne sa môžu upravovať v moduly X20 umiestnené v „PhysicalView“. Podrobný popis všetkých parametrov je k dispozícii v sekcii "Standard function model" X20 modul krokového motora „datasheet“

Pre náš konkrétny motor necháme pôvodné hodnoty. Pretože nevieme ako sa motor bude správať v reálnej aplikácii. Tieto hodnoty zmeníme neskôr keď budeme testovať pripojenie a krokový motor.

Po týchto nastaveniach môžeme pokračovať k vytvoreniu CompactFlesh, na ktorú sa nahrajú údaje ako je nastavenie pripojenia, nastavenie verzie Runtime, knižnice, a hardvérové vybavenie ktoré bolo pripojené.

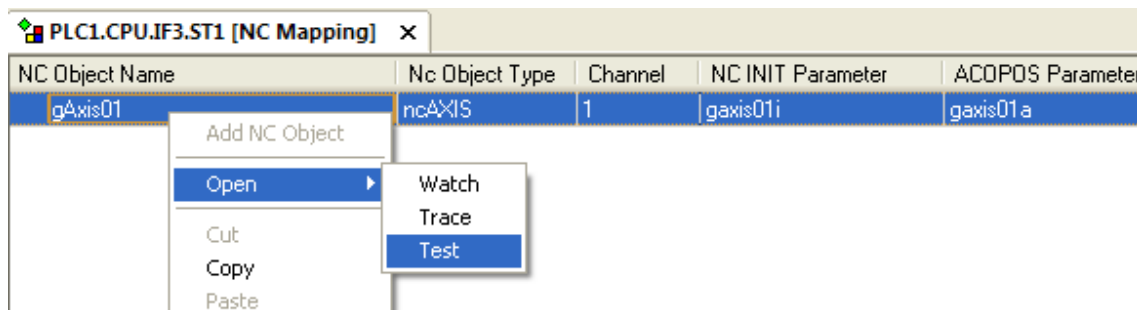
Systém nabootuje informácie s CF. Po dokončení operácie sme schopný sa pripojiť k zariadeniu on-line. Ak v dolnom paneli svieti RUN sme pripravený pracovať so zariadením. Panel obsahuje aj informácie o názve PLC ktoré je pripojené, verziu Runtime, ktorú zariadenie používa a IP adresu.

INA: tcpip/DAIP=192.168.0.10 /REPO=11159 /ANSL=1	X20CP1484 U3.09	RUN
--	-----------------	-----

Obr. 5.4 Ukazovateľ spodnej lišty v Automation studio.

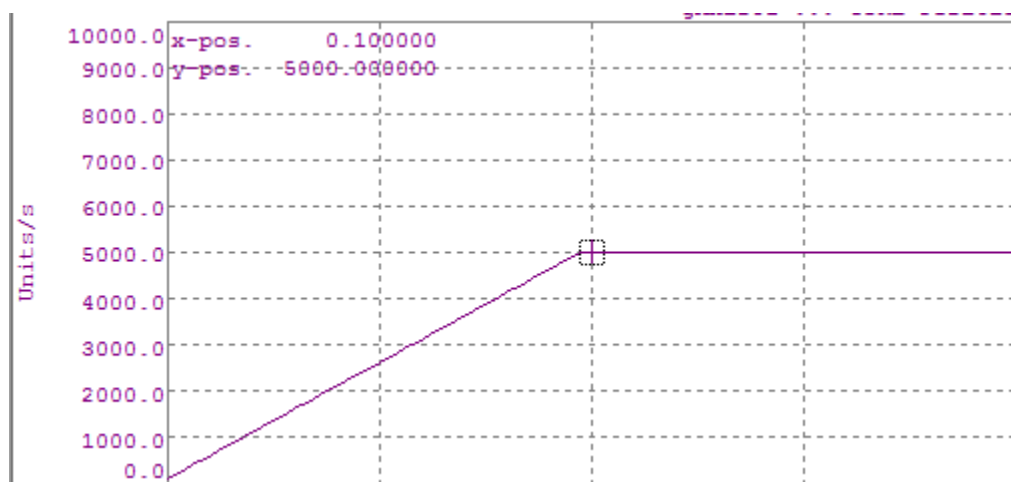
5.2 PROGRAMOVANIE JEDNEJ OSI

Pre kontrolu správneho pripojenia a nastavenia je vhodné spustiť funkciu Test.



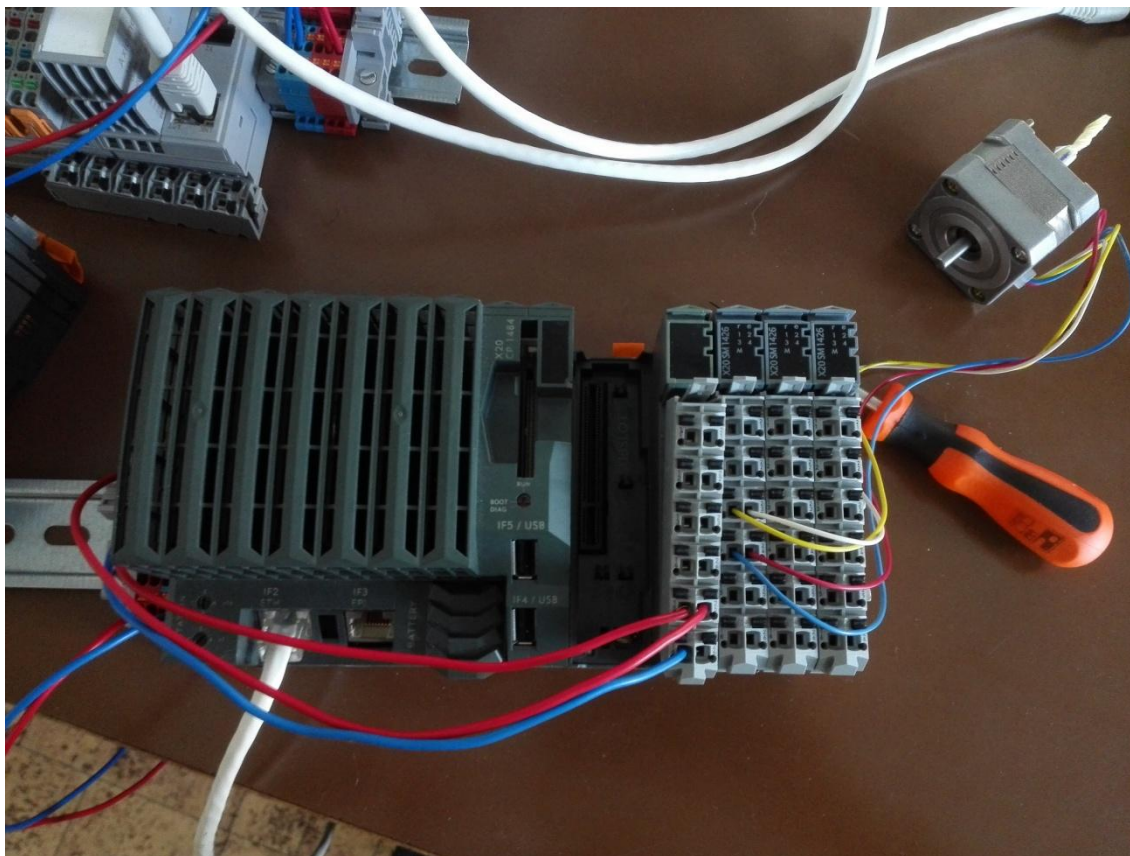
Obr. 5.5 Spustenie funkcie Test.

Funkciou si overíme či motor spolupracuje s riadiacim modulom. Umožňuje nám meniť nastavenia prúdov počas testu čím sme schopný nastaviť ideálne hodnoty, kedy bude motor maximálne využitý a systém nebude vyhadzovať error. Počas testu sme postupným zvyšovaním/znižovaním nominálnej hodnoty prúdu prišli na to, že motor najlepšie pracuje pri 50% nominálnom prúde čo je 0.5A, pri vyšších hodnotách dochádzalo k erroru a prekročeniu zadanej hodnoty posuvu. Maximálny prúd sme preto nastavili na 70% a zádržný na 15%. Z funkcie sme dostali graf ktorý ukazuje ako rýchlo motor dosiahne navolenú rýchlosť. Z grafu je vidieť, že motor dosiahne zvolenú rýchlosť za 0.1 sekundy.



Obr. 5.6 Graf závislosti rýchlosti na čase.

V aplikácii je možnosť overenia presnosti polohovania, kedy si nastavíme referenčnú hodnotu a hodnotu posuvu. Pri tomto overovaní polohovanie dosiahlo maximálnu presnosť. Program v PLC obsahuje predvolené jednotky PLCunits v našom prípade 1000PLCunits = 1otáčka rotoru. Tieto parametre sme schopný v programovej časti prepočítať na mm alebo inú jednotku v sústave SI.

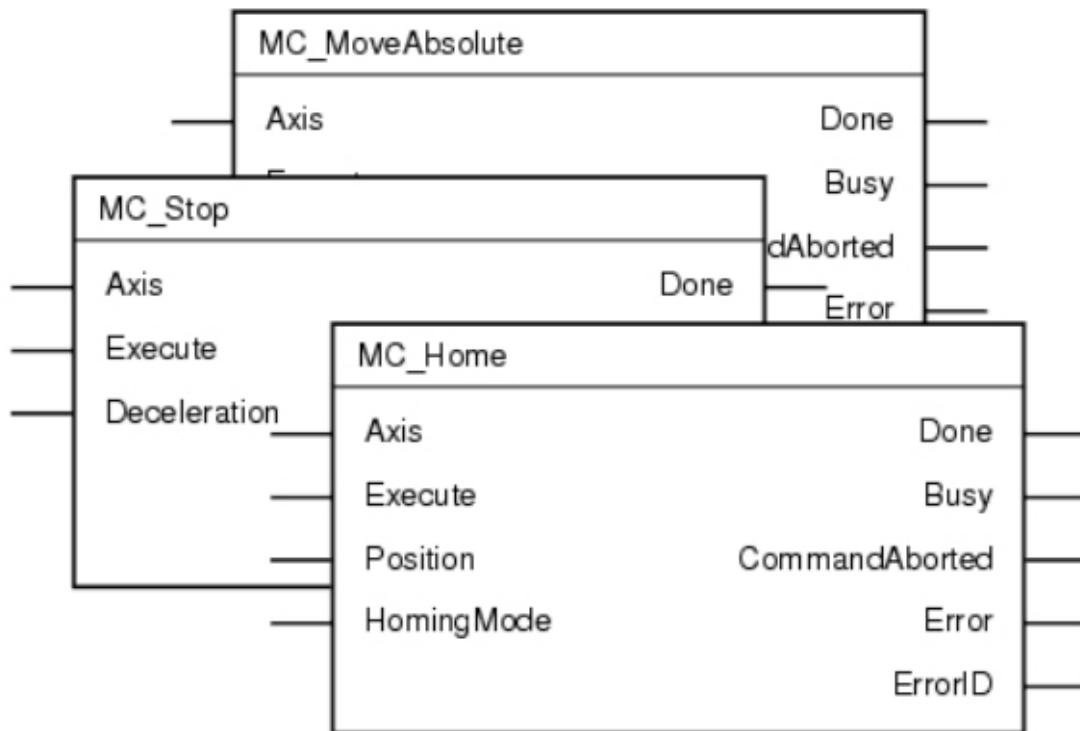


Obr. 5.7 Zapojenie počas testu.

5.2.1 ACP10_MC - PLCopen MOTION CONTROL LIBRARY

Je knižnica v Automation studio ktorá bola použitá pre polohovanie manipulátora. Okrem funkcie PLCopen¹⁰, knižnica ACP10_MC tiež obsahuje pokročilé funkcie, bloky, ktoré implementujú funkcie týkajúce sa systémov B & R. Toto umožňuje použiť jednoduché pohonné funkcie ako základ pohybov a prípravu pohonov, ale aj náročnejšie funkcie ako je implementácia pohybov včkových profilov.

¹⁰PLCopen je organizácia aktívna v priemyselnom riadení, vytvára vyššiu účinnosť vo vývoji aplikačného softvéru a znižuje náklady. Táto spoločnosť má veľkú zásluhu na nezávislosti hardvéru od softvérového kódu.



Obr. 5.8 Funkčné bloky s knižnice ACP10_MC. [1]

Väčšina funkčných blokov má tieto vstupné a výstupné parametre ako môžeme vidieť na obr.5.8:

- | | |
|-----------------------------|---|
| • Axis (vstupný) | – odkaz na osu, s ktorou funkcia pracuje. |
| • Execute/Enable (vstupný) | – vstupná podmienka pre zahájenie akcie. |
| • Busy (výstupný) | – akcia sa momentálne robí. |
| • Done (výstupný) | – úspešné ukončenie akcie. |
| • CommandAdorted (výstupný) | – akcia bola prerušená. |
| • Error (výstupný) | – chyba pri vyvolávaní funkčného bloku. |
| • ErrorID (výstupný) | – číslo chyby. |

PLCopen motion control definuje osem základných stavov, v ktorých sa os môže nachádzať. Jednotlivé stavy sú definované takto:

- | | |
|--------------------|---|
| • Disable | – pohon je vypnutý. |
| • StandStill | – pohon je zapnutý, ale nevykonáva žiadny pohyb. |
| • Homing | – prebieha nastavovanie referenčnej polohy. |
| • ErrorStop | – nastala porucha pohon sa nepohybuje. |
| • Stopping | – zastavenie pohybu. |
| • DiscreteMotion | – os vykonáva diskretný pohyb (bude sám ukončený). |
| • ContinuousMotion | – os vykonáva spojitý pohyb (ukončí sa až určitou udalosťou). |

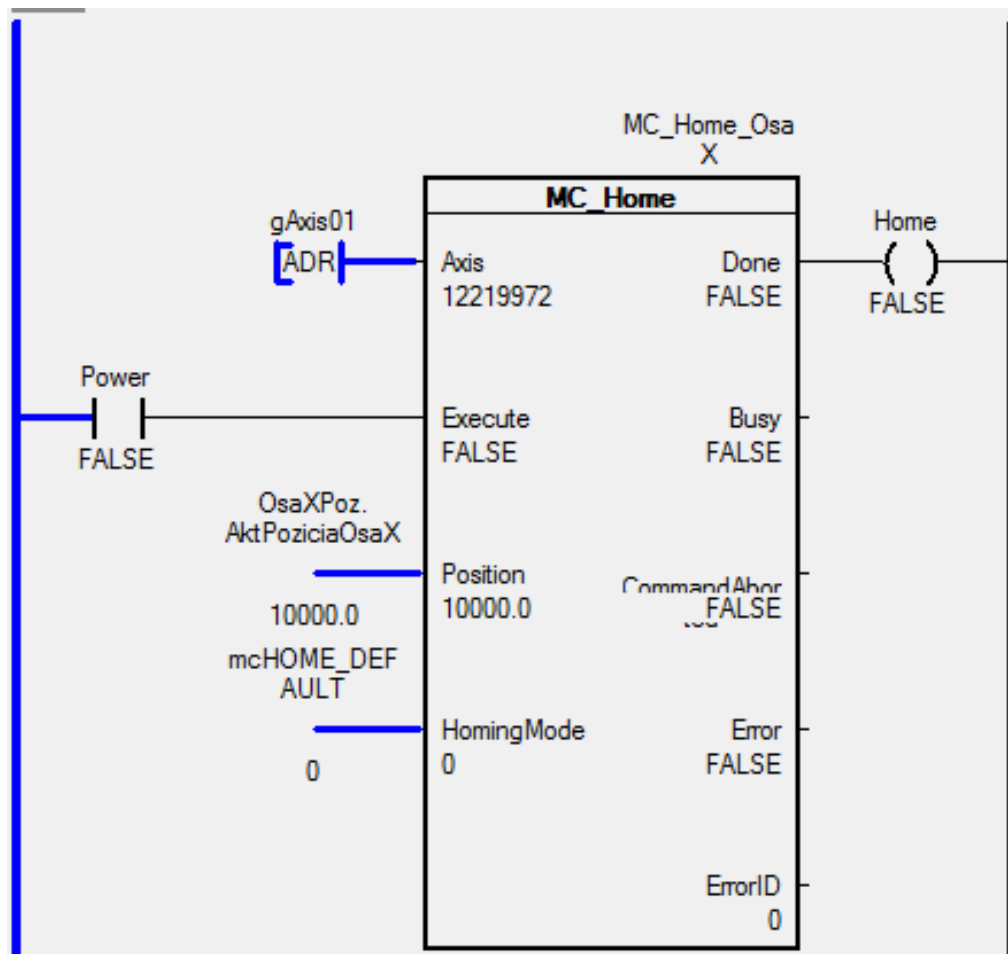
PRAKTICKÁ ČASŤ

PLCopen má zadefinované veľké množstvo funkčných blokov pre prácu s osou/osami. Uvedieme si tie najzákladnejšie s nich:

- MC_Power – zapnutie pohonu.
- MC_Home – nastavenie referenčnej hodnoty osi.
- MC_MoveAbsolute – pohyb na určenou polohu.
- MC_MoveAdditive – pohyb o určitou vzdialenosť.
- MC_Stop – zastavenie pohybu motoru.
- MC_ReadActualPosition – čítanie aktuálnej polohy osi.
- MC_ReadActualVelocity – čítanie aktuálnej rýchlosti osi.
- MC_ReadActualTorque – čítanie aktuálneho točivého momentu.
- MC_ReadAxisError – čítanie chybových stavov.
- MC_ReadStatus – čítanie statusu osi (servo zosilňovače).
- MC_Reset – reštart chybového stavu.

Program bol vytvorený v programovacom jazyku Ladder Diagram (LD). Do programu boli vložené funkčné bloky pre polohovanie motora v jednej osi. Funkčný blok je programový blok, ktorému je vyhradená časť pamäte. Pri každom volaní funkčného bloku je potrebné priradiť hodnoty alebo premenné na vstup do funkčného bloku. Inicializačná časť programu, v ktorej sa zadávajú požadované parametre bola vytvorená pomocou jazyka Structured Text (ST). V tejto časti sa nachádzajú tieto parametre: vzdialenosť (posuvu), rýchlosť, zrýchlenie a spomalenie. Všetky parametre majú preddefinovanú jednotku PLCunits. Na obr. 5.9 je vidieť funkčný blok MC_Home, v ktorom sú informácie o kludovom stave pohonu. Position predstavuje aktuálnu pozíciu, v ktorej sa os nachádza. HomingMode predstavu referenčnú pozíciu manipulátora. V obr. 5.10 je znázornená inicializačná časť, v ktorej sa zadávajú požadované parametre.

Po pripojení PLC k motorom manipulátora bolo vhodnejšie použiť pre polohovanie jednotky SI. Preto bola v program vytvorená nová premenná ktorá sa zadáva v μm . Jednotka μm bola zvolená s toho dôvodu, že dopravník má relatívne malý rozsah posuvu. Na hriadeľ motoru je pripojená závitová tyč M6x0,5. S týchto informácií si vieme vyjadriť posuv. Ak motor urobí jednu otáčku dopravník sa posunie o 0,5mm. Pri testoch presnosti polohovanie bolo použité posuvné meradlo s presnosťou 0,05mm. A teda presnosť polohovanie bolo 0,05mm.



Obr. 5.9 Ukážka s programu.

```

PROGRAM _CYCLIC
(* Jednu otacka = 1000 PLCunit, dopravnik je polohovany
pomocov zavitovej tyce ss stupanim 0,5mm s toho vypliva,
ze 2000PLCunits = 1mm posuvu, zadavanie polohy budeme zadavat
v mikrometroch kvoli tomu, ze maximalne posuvy su relativne male
2000PLCunits = 1000 microm = 1 mm *)
Svmicrom:=5650;
Szaporna:=- (Svmicrom*2);
Skladna:=(Svmicrom*2);

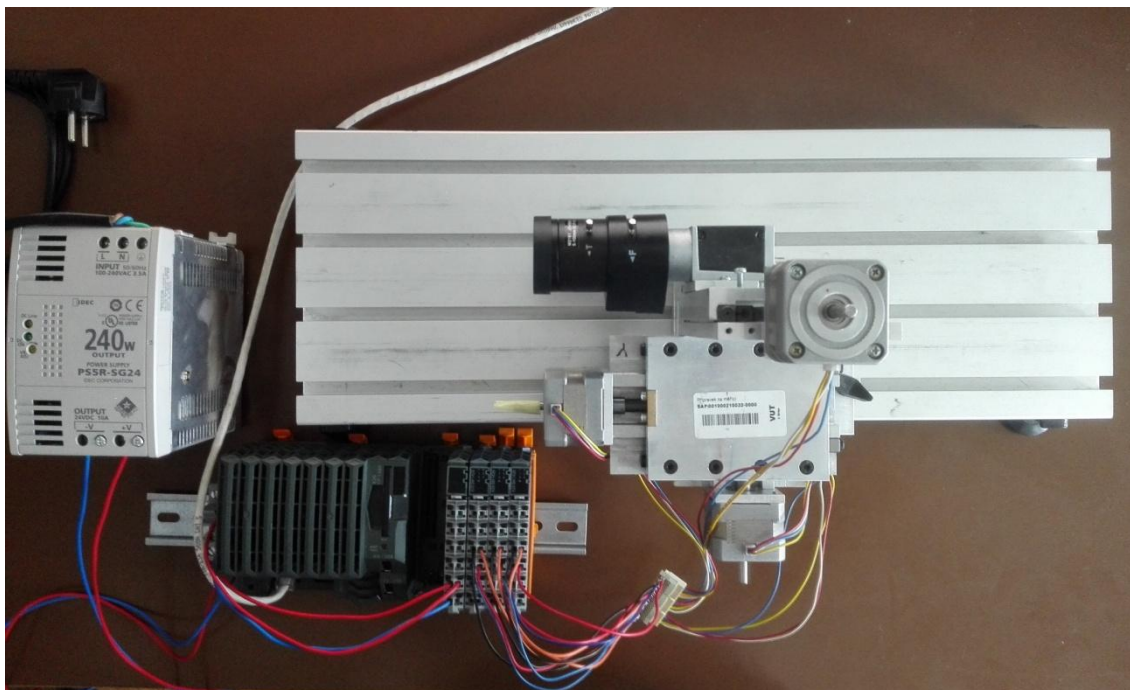
OsaXParametre.VzdialenostKladna :=Skladna;
OsaXParametre.VzdialenostZaporna :=Szaporna;
OsaXParametre.Rychlost :=5000;

OsaXParametre.Zrychlenie := 10000;
OsaXParametre.Spomalenie := 5000;

END_PROGRAM
    
```

Obr. 5.10 Inicializačná časť zdrojového kódu.

5.3 PROGRAMOVANIE CELÉHO MANIPULÁTORA



Obr. 5.11 Fotografia manipulátora.

Pri programovaní všetkých troch osí sa vyskytol problém z nedostatočnou pamäťou CPU. Inteligentné riadenie pomocou knižnice ACP10_MC je veľmi náročné na pamäť. Zapožičané PLC má malú pamäť pre takéto riešenie pohonu. Preto musela byť využitá alternatívna možnosť riadenia všetkých osí a to pomocou Ramp módu.

Ramp mód využíva štruktúru stavového automatu [8], kedy sa používajú ControlWord – pre riadenie stavov a StatusWord – pre čítanie aktuálneho stavu na výstupe karty. Stavový automat postupne prechádza z jedného stavu do druhého ako je znázornené na stavovom diagrame na obr. 5.13. Na prechod do nového stavu potrebuje aby bol splnený predchádzajúci stav, to sa vyhodnocuje podľa vstupnej premennej ktorá obsahuje Status word. Bitové hodnoty premenných podľa aktuálneho stavu sú zapísané v tabuľke 5.1.

Za týmto účelom bol vytvorený nový funkčný blok ktorého úlohou je správať sa ako stavový automat. Zdrojový kód pre manipuláciu dopravníkom bol zostavený s ktoré najskôr overovali StatusWord získavané zo vstupu karty (MpGenStatus) a následne podľa stavu v ktorom sa automat nachádzal priradzovali adekvátnu hodnotu stavu podľa ControlWord na výstup karty (MpGenControl). Postupnými cyklami sa automat dostal do stavu Operation enable kedy sa mohlo pristúpiť k inicializácii. Funkčný blok pozostáva s troch vstupných premenných, dvoch výstupných a jednej vstupno/výstupnej premennej. Pre ovládanie viacerých motorov bez nutnosti vytvárania ďalších funkčných blokov, je potrebné vytvoriť premennú pre každý motor zvlášť. Pre dané riešenie boli vytvorené tri premenné MotorX, MotorY a MotorZ. Daným premenným bol priradený funkčný blok pod názvom StepMotor. Každá premenná sa správa ako samostatný

PRAKTICKÁ ČASŤ

funkčný blok, preto sa motory môžu riadiť úplne nezávislé na sebe alebo všetky súčasne. To záleží na rozhodnutí užívateľa. Na základe daných faktov bola jedna premenná priradená práve jednej karte. Premenná MotorX ako aj ostatné má v sebe vstupy ako aj výstupy s funkčného bloku. Preto sme schopný priradiť adekvátny vstup s funkčného bloku na adekvátny vstup karty a to isté platí aj pre výstupy. Na obr. 5.12 je znázornené mapovanie premenných na kartu. Jedna premenná funkčného bloku ktorá je vstupnou a zároveň výstupnou pod názvom Enable, sa využíva pre vstup do funkčného bloku a následne jeho spustenie, pričom nám dáva na výstupe informácie o tom, či je motor zapnutý a pripravený na použitie. V programe cyclic prebieha volanie funkčného bloku pre každý motor. Pri každom volaní funkčného bloku musia byť na jeho vstupy priradené aktuálne hodnoty ktoré medzi časom nastali, čo je zabezpečené čítaním vstupov karty a následným zapisovaním na vstup funkčného bloku.

Karty X20SM1426 sú vybavené výstupmi na ovládania krokových motorov ale taktiež v sebe majú štyri digitálne vstupy DI 1...DI 4 ako je zobrazené na Obr.3.4. Všetky štyri digitálne vstupy sú namapované na jeden vstup karty pod názvom InputStatus, ten má dátový typ USINT čo znamená, že jeho veľkosť je 8bitov/1byte alebo 0 až 255. Ako v kapitole 5.3 aj tu sa stretávame s jednotkami PLCunit preto je možné použiť inicializačnú časť s predchádzajúcej úlohy ako je na obr.5.10. Posuv manipulátora je ovládaný pomocou štyroch prepínačov. Prvý, ModuleStart, nám spúšťa/zastavuje celú činnosť, tento prepínač je namapovaný na premennú Enable ktorá je spomínaná vyššie, informácie o zapnutí či vypnutí posielajú na vstup karty cez DI 2, bitová hodnota pre zapnutie je 0000_0010 a pre vypnutie 0000_0000. Zvyšné tri prepínače, bfX, bfY, bfZ, sú namapované na každú kartu zvlášť cez DI 1 bitová hodnota pre kladný posuv je 0000_0001 a pre záporný posuv je 0000_0000. V programe sme schopný pristupovať na každý bit samostatne a to tak, že za premenu sa priradí číslo adekvátneho bitu ktorí chceme sledovať. Tieto prepínače nám poslúžia na zmenu polohy všetkých troch osí s kladnej na zápornú a naopak, pričom každá je ovládaná samostatne a na sebe nezávislé.

➔ AbsPos01	Program:MotorZ.ChcenaPozicia	DINT
➔ MpGenControl01	Program:MotorZ.ControlWord	UINT
➔ MpGenMode01	Program:MotorZ.ModPohybu	SINT
➕ AbsPos01ActVal	Program:MotorZ.AktualnaPozicia	DINT
➕ MpGenStatus01	Program:MotorZ.StatusWord	UINT
➕ InputStatus	Program:bfZ	USINT

Obr. 5.12 Popisuje priradenie premenných na I/O karty

PRAKTICKÁ ČASŤ

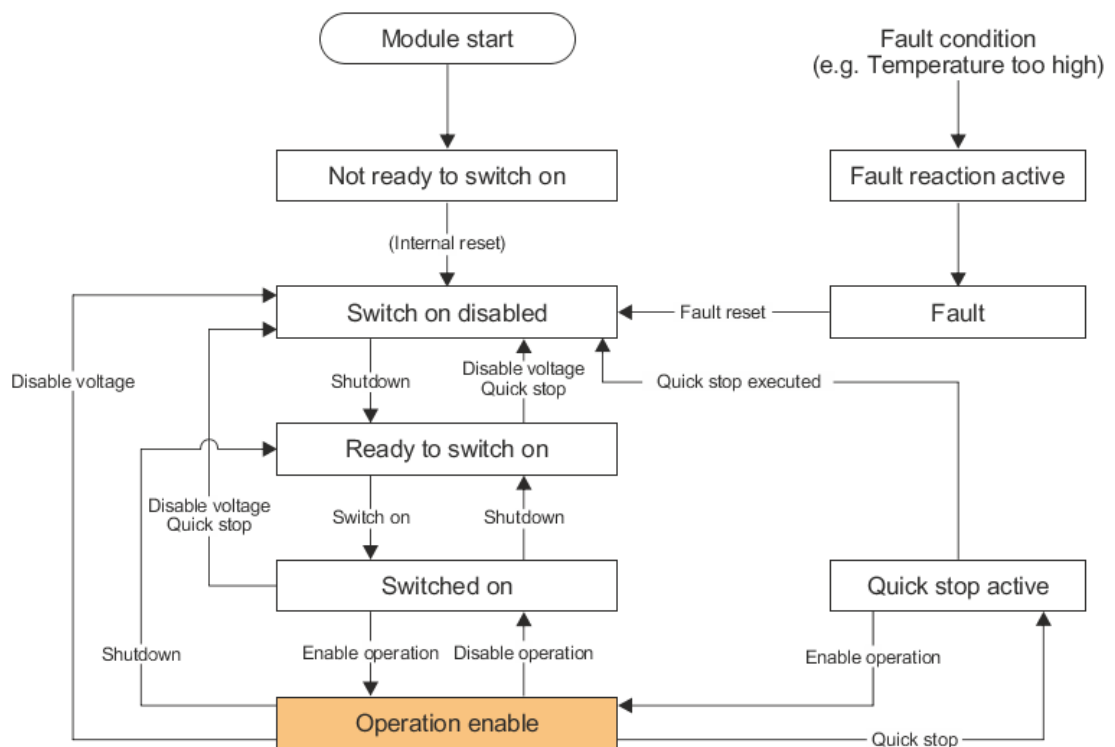
Tabuľka 5.1 Bitové hodnoty prislúchajúce stavom. [1]

Control word

Command	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit ⁽¹⁾																
Shutdown	x	x	x	x	x	x	0	x	0	x	x	x	x	1	1	0
Switch on	x	x	x	x	x	x	0	x	0	x	x	x	0	1	1	1
Disable voltage	x	x	x	x	x	x	0	x	0	x	x	x	x	x	0	x
Quick stop	x	x	x	x	x	x	0	x	0	x	x	x	x	0	1	x
Disable operation	x	x	x	x	x	x	0	x	0	x	x	x	0	1	1	1
Enable operation	x	x	x	x	x	x	0	x	0	x	x	x	1	1	1	1
Fault reset	x	x	x	x	x	x	0	x	↑	x	x	x	x	x	x	x

Status word

Status	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit																
Not ready to switch on	x	x	x	x	x	x	1	0	x	0	x	0	0	0	0	0
Switch on disabled	x	x	x	x	x	x	1	0	x	1	x	0	0	0	0	0
Ready to switch on	x	x	x	x	x	x	1	0	x	0	1	0	0	0	0	1
Switched on	x	x	x	x	x	x	1	0	x	0	1	1	0	0	1	1
Operation enable	x	x	x	x	x	x	1	0	x	0	1	1	0	1	1	1
Quick stop active	x	x	x	x	x	x	1	0	x	0	0	1	0	1	1	1
Fault reaction active	x	x	x	x	x	x	1	0	x	0	x	0	1	1	1	1
Fault	x	x	x	x	x	x	1	0	x	0	x	0	1	0	0	0



Obr. 5.13 Stavový diagram pre funkciu Ramp mód. [1]

6 ZÁVER

Hlavným cieľom práce bolo navrhnuť program pre polohovanie manipulátora, ktorý na základe inicializačnej časti v programe urobí posuv v danej osi o zadnú hodnotu posuvu, rýchlosti, zrýchlenia a spomalenia. PLC prečíta nové hodnoty polohy a zapíše ich späť do programu. Týmto riešením sa predchádza prekročeniu maximálneho posuvu, a tým zabezpečenie pred kolíziou. Ako hlavný nástroj pre túto prácu bol využitý hardvér a softvér spoločnosti B&R. Prvým krokom programovania je zvolenie správnych komponentov a zaradenie ich do programu, čím vznikla správna komunikácia medzi PLC, PC a motormi.

Druhou fázou bola konfigurácia modulu pre riadenie motorov. V tejto časti sa testovali a nastavovali hodnoty prúdu a rýchlosti. Testovanie bolo časovo náročné pretože systém vyhodil chybu pri každej nezrovnalosti, ako bola veľkosť prúdu, veľkosť zrýchlenia, maximálna rýchlosť. Po každej systémovej chybe sa musel systém reštartovať. Napokon sa podarilo nastaviť optimálne hodnoty prúdu pre plynulý rozbeh motora, bez rázov alebo zasekávania čo stišlo prevádzku a s kultivovalo chod motora. V prípade, že bol prúd nastavený na takmer maximálnu hodnotu, motor vykazoval pri rozbehu rázi. Naopak keď bola hodnota pri nízka motor sa buď nerozbehol vôbec, alebo prerušovane. Tieto isté podmienky platili aj pri nastavovaní rýchlosti, zrýchlenia a spomaľovania.

S využitím knižnice PLC10_MC a jej funkčných blokov sa naprogramovalo riadenie jednej osi a overovanie presnosti polohovania. Na overenie presnosti polohovania sa využíva funkčný blok ktorý číta aktuálnu polohu. Porovnaním zadanej a aktuálnej polohy sme schopný overiť dosiahnutú presnosť. Počas práce sme sa stretli s niekoľkými chybami, či už to bola zle nastavená hodnota alebo zle priradenie premennej. Keďže práca je vykonávaná na bežnom PLC je problém z menením hodnôt v inicializačnej časti programu za spustenej prevádzky. Program sa musí vždy zastaviť a po zmene údajov znova nahráť. Tento problém by sa dal vyriešiť použitím OPLC. Kedy by sa zadávali nové požiadavky pre posuv priamo na displej.

Presnosť polohovania bolo overovaná ako experimentálne (pomocou posuvného meradla), tak aj softvérovo. V prípadoch kedy bolo použité posuvné meradlo polohovanie vyšlo s presnosťou 0,05mm.

Po úspešnom ovládaní jednej osi sme sa presunuli k riadeniu celého manipulátora. Pomocou troch motorov. Os X, Y, Z teda riadenie manipulátora v trojrozmernom priestore. Kvôli nedostatočnej pamäti bola použitá náhradná funkcia pre riadenie Ramp mód, pomocou tejto funkcie sa polohovanie všetkých osi podarilo takmer bez problémov. Základným princípom bolo pochopenie ako celý stavový automat funguje a následne jeho funkcionality vložiť do funkčného bloku. S funkčného bloku sa stal stavový automat. Vďaka tomu sme sa mohli naplno venovať polohovaniu všetkých osí.

7 ZOZNAM POUŽITÝCH ZDROJOV

- [1] <https://www.br-automation.com/en/perfection-in-automation/>
- [2] Shell, R.L., Hall, E.L.: Handbook of industrial automation. New York: M. Dekker, c2000, xii, 900 p. ISBN 0-8247-0373-1
- [3] <http://www.promotic.eu/cz/pmdoc/Subsystems/Comm/PLC/BerneckerReiner.htm>
- [4] <https://www.myodesie.com/wiki/index/returnEntry/id/2962>
- [5] http://www.plcdev.com/stage_programming_tutorial_by_example
- [6] <http://automatizace.hw.cz/co-se-skryva-pod-oznacnim-plc>
- [7] <http://www.plc-automatizace.cz/knihovna/historie/historie-plc.htm>
- [8] <http://www.drivesandsystems.com/plc-program-sequences/>
- [9] Bennett, S. (1993). A History of Control Engineering 1930-1955. London: Peter Peregrinus Ltd. On behalf of the Institution of Electrical Engineers. ISBN 0-86341-280-7
- [10] <http://embedded-computing.com/eletter-products/iec-61131-3-plc-software-programming-and-runtime-solution/>
- [11] <https://www.lifewire.com/what-is-ethernet-3426740>
- [12] <http://www.systec-electronic.com/en/systec-engineering-development-services/automation/embedded-sensors-and-small-devices/plc-programmable-embedded-controller-with-ethernet-can-canopen-and-modbus>
- [13] <http://www.ni.com/white-paper/3367/en/>

8 ZOZNAM POUŽITÝCH SKRATIEK A SYMBOLOV

PLC	– Programovateľný logický automat
PC	–ProgrammableController
SPS	–SpeicherProgrammierbareSteuerung
FPC	– Free ProgrammableController
PA	–Programovateľný automat
OPLC	– Programovateľný logický automat so zabudovaným panelom
NEMA	– Národná elektrikárska asociácia výrobcov
PC	– Počítač
CPU	– Centrálna procesorová jednotka
I/O	– vstupy / výstupy
DI	– Digitálne vstupy
DO	– Digitálne výstupy
AI	– Analógové vstupy
A/D	– Analógový / digitálny prevodník
AO	– Analógové výstupy
D/A	– Digitálny / analógový prevodník
IP67	– Stupeň ochrany proti prachu a vode
B&R	– Bernecker&Reiner
CF	– Compact Flash
USB	– Univerzálna sériová zbernica
U [V]	– Napätie
I [A]	– Prúd
LD	– Ladder Diagram
ST	– Structured Text
SDM	– Systém diagnostik manažér
RAM	– Pamäť
CNC	– Číslicovo riadený stroj
s[m]	– Posuv

9 ZOZNAM OBRÁZKOV A TABULIEK

<i>Obr. 3.1 Popis PLC firmy B&R.[1]</i>	15
<i>Tabuľka 3.1 Rozhrania prislúchajúce PLC X20CP1484. [1]</i>	16
<i>Obr. 3.2 Popis zapojenia napájania pre PLC. [1]</i>	17
<i>Obr. 3.3 Význam jednotlivých kontroliek na LED panely.[1]</i>	17
<i>Obr. 3.4 Príklad zapojenia modulu s krokovým motorom a senzormi. [1]</i>	18
<i>Obr. 3.5 Úvodná obrazovka s diagramom krokov pri práci. [1]</i>	19
<i>Obr. 3.6 Pracovné prostredie. [1]</i>	20
<i>Obr. 3.7 Graf cyklov a umiestnenie podľa priority operácie. [1]</i>	22
<i>Obr. 4.1 Schéma riadenia pohybu. [1]</i>	23
<i>Obr. 4.2 Základné komponenty použitého systému.</i>	24
<i>Obr. 5.1 Strom pripojených zariadení.</i>	26
<i>Obr. 5.2 Reálne zobrazenie PLC s pridanými modulmi (prevzaté so System Designer).</i>	26
<i>Obr. 5.3 Nastavenie parametrov motora.</i>	27
<i>Obr. 5.4 Ukazovateľ spodnej lišty v Automation studio.</i>	28
<i>Obr. 5.5 Spustenie funkcie Test.....</i>	29
<i>Obr. 5.6 Graf závislosti rýchlosti na čase.....</i>	29
<i>Obr. 5.7 Zapojenie počas testu.</i>	30
<i>Obr. 5.8 Funkčne bloky s knižnice ACP10_MC. [1]</i>	31
<i>Obr. 5.9 Ukážka s programu.....</i>	33
<i>Obr. 5.11 Fotografia manipulátora.....</i>	34
<i>Obr. 5.12 Popisuje priradenie premenných na I/O karty.....</i>	35
<i>Tabuľka 5.1 Bitové hodnoty prislúchajúce stavom. [1]</i>	36
<i>Obr. 5.13 Stavový diagram pre funkciu Ramp mód. [1]</i>	37